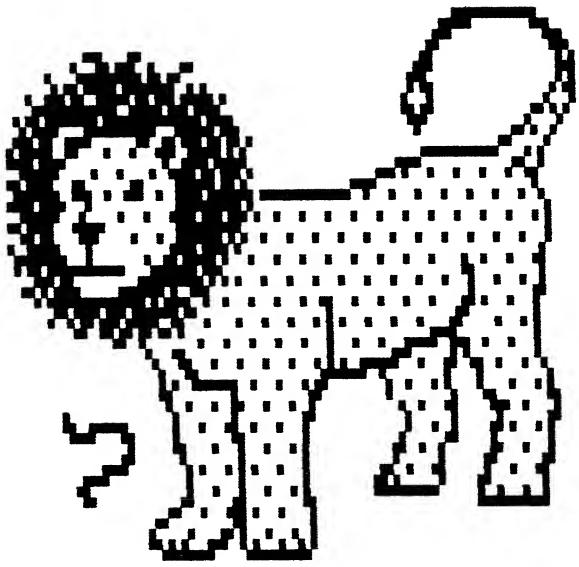
ATARI
COMPUTER
ENTHUSIASTS
3662 Vine Maple Dr. Eugene OR 97405



THE ST'S ARE HERE

BUMPAS REVIEWS

Well, I sold my 800. It was a real emotional experience. After all, it was my first computer. I learned most of what I now know on the machine. And it never gave me any trouble. It was never down during the four years I owned it. The IBM I own at work suffered two complete breakdowns in the first year after the warranty expired. The repair costs on each occasion could have paid for an entire Atari system. Of course I only got a fraction of the $1100 I paid for the 800.

Now I own a 130 XE! It's great! I believe the keyboard alone has boosted my typing speed by more than 10%. And I type 75 wpm the last time I tested myself on the 800. I type so fast on the XE that a character or two in each paragraph never gets to the screen when I use PaperClip. I guess it has so many features it doesn't have time to check the keyboard buffer often enough to keep up with me! Writer's Tool has no problem though. If you don't type fast, PaperClip might provide you with the best price/performance ratio, because it does have a lot of features.

The DOS 2.5 disk the club distributes has several utility files of interest. Of course there is RAMDISK.COM, which automatically installs a 64k ramdisk (it's D:8!). Then there is a program to convert DOS 3 files. Another program automatically creates an autoboot disk from a screen menu (I like this one). And there is a configuration program which permits you to make additional disk assignments when you have more than 2 drives.

One obvious omission is a means to reassign the ramdisk to something other than D:8. Many programs are designed to access only 2, or 4 drives. If a user could assign the ramdisk to any number, say from 2 through 8, the flexibility of this ramdisk would be increased many times. I know one user has already patched the system to assign the ramdisk to D:4. I won't mention his name so no pressure will be put on him to write a short little article describing the patch we need to make. But maybe someone out there will come up with an elegant solution to this problem. Any takers?

My 130 XE has an accidental feature which makes it a "one-off", sort of like that postage stamp with the airplane printed upside down. My Control Key has the word "Shift" printed on it, just like the larger Shift Key below it. So far, I've found no other machine with this characteristic. If you know of another XE machine with "Shift" printed on the Control Key, let me know, I'm keeping count!

So far, the XE runs everything I've tried, except M.U.L.E. I've heard some other Electronic Arts stuff (One on One) won't run either. Lucky I added a 64k board to my 400, so I can still play M.U.L.E.! The XE is so small, I have it along with my 400 in the same space of my computer cabinet which was formerly occupied by the 800. The XE is actually smaller than the 400, although it's a little bit wider. And with the power switch in the rear, my son Aden hasn't figured out how to turn off the machine just before I've saved a 100 sector file for the newsletter!

OPERATION MARKET GARDEN (SSI, $50) is an operational level simulation of the drive on Arnhem in September, 1944 which also involved the multi-divisional air assault. It's for one or two players, and the map takes up most of a 32x31 hex grid containing 13 types of terrain. Ten different types of units are portrayed in the game.

Player options also include 4 levels of play, historical or free set up, historical or random weather effects, hidden units, and intermediate or advanced play. Advanced play permits additional options and creates additional difficulties. For instance, supply is more difficult to manage. But you may order advances after combat and execute offensive and protective artillery barrages. Stacking is limited and Allied engineers find it more difficult to build bridges. Command control may be lost in the advanced game. Advanced players may not examine enemy units.

Game operations include air strikes which may interdict movement, or may attack ground units. Units may be built up or broken down into component units. Each unit's activity expends "operation points". When the points are gone, no more orders are given to the unit.

Game functions are controlled entirely with the keyboard using one or more of the half-dozen menus provided in the game. A 16-page manual describes all the game functions and includes hints on play and a brief history of the battle. Appendices include 3 historical maps, a page of tables and abbreviations used, order of appearance for the units in the game, and a three page order of battle. Two plastic laminated maps with charts and tables on the reverse side are provide as play aids. The maps are full-color and a welcome addition to the game for plotting strategy.

This one is tougher to win solitaire than most computer simulations. You'll be busy with this one for quite awhile.

SIX-GUN SHOOTOUT (SSI, $40) is a simulation which has not been successfully attempted before. It's the wild west, and one or two players may choose from among 18 weapons and maneuver around up to 25 types of terrain. There are ten scenarios which include historical ones like the Gunfight at the OK Corral. There are also 3 scenarios from movies, like The Good, The Bad and The Ugly. Two "typical" scenarios include townspeople vs. trailhands and Indians vs. "Travelers".

The scenarios provide a player with from 2 to 18 fighters for a total of up to 30 fighters on screen at one time. You can save a game in progress for later play. The menu options actually allow for more than 10 scenarios as players can take a scenario as it comes, or they can choose to randomize the starting position and characteristics of the fighters.

Commands are entered with the keyboard and include the ability to show the map with all the characters removed, and to paint orange squares over the entire area any given character can see from the ground. When operating your characters, you can change weapons and load them. You can move, fire them, or fight hand to hand. Characters can stand, kneel, or lie prone (for better cover, but your movement is limited). You can jump through windows, hide behind walls and use other terrain. One character can even throw dynamite!

The program gets a little gruesome sometimes. Occasionally, you are offered the choice of shooting a character in the legs, the gut, or the chest! A big feature of the program is the option to create a personal character for the player. This character starts with low characteristics, but your character appears in each scenario and offers you the opportunity to add skill points and become a very powerful character — maybe even more powerful than Billy the Kid!

The 24-page manual not only describes the game functions fully, but also includes detailed scenario descriptions and historical notes and play aid charts. If you like western movies, here's your chance to get into the action yourself!

ON TRACK (by GameStar, and distributed by Activision $35?) is described as "computer model car racing". And it is!

When I was in high school, several of my friends had slot cars which they raced several times a week. These slot cars were expensive. And to race them you needed a rather substantial slot car track. Few were rich enough to have their own track, so stores set them up and charged fees to racers. **On Track** is a good simulation of these slot car racers. The cars even spin out when you lose control, just like the slot cars.

In this GameStar release you have 10 tracks from which to choose (20, if you count the ability to surface each track with your choice of dirt or pavement!). One or two players may play, but I suggest you make it two player until you get very good. The Atari embarrases me so far. The player may also choose one of three characters (drivers) provided in the program. Each driver is provided with characteristics which affect his driving skill and qualities of the vehicle he drives. The variations do not seem to be oppressive. The player's skill is much more important.

The joystick is the controller. Left and right steers the car. Forward and backward shifts gears between Lo and Hi. The button is the brake. Menu options permit gear shifting to be automatic (a nice touch). I use "automatic" brakes, also, because I haven't yet developed the skill to do correct braking. Luckily, when you run your car off the track, it doesn't crash. It comes to a stop. If you just sort of graze the edge of the track, you can slow down or at least get started quickly again. Well, to confess, I really drive like a blind man. I run until I hit something. Then I change direction. But I have seen a player sweep smoothly around those turns, and it's exciting!

Game options also include racing for the fastest time, or for the greatest distance.

I've left the best feature for last: Both players' cars are on the same screen for the whole race! For the slot car idea this is perfect. No split screens, or scrolling. You can see at all times how the race is going. You can take the car across country (you don't have to stay in the "slots"). I tried this, but when you cross the start line you don't get credit for the lap.

This is the most exciting racing game for two players I've yet seen. You'll have many hours of fun with this one.

— Jim Bumpas
Co-Editor

NEWS AND REVIEWS

by Mike Dunn, Co-Editor

The big excitement now is the awaiting of the new ST's. A number of our members have ordered theirs (brave souls!) and they are supposed to be here "Real Soon Now", or next week. The first shipments are going to user groups to help de-bug the machines. Speaking of the ST's, some Macintosh programs are almost ready for the ST. Mosaic Software's **Twin**, a Lotus 1-2-3 type program, has been announced to be included with the 520ST, or $99 alone. VIP Technologies are releasing another 1-2-3 clone, **The Professional** which will be for a number of computers, including the 520 ST. This issue is our July/Augest issue, so hopefully a full report on a user's experiences with the ST's will be in the next issue (or maybe even sooner).

The last local meeting of ACE was another great one, with Ralph Walden, author of **ACE-C**, giving an impressive demo of his new C program for the XE, which has a built in Editor which does syntex checking, and, with a custom RamDisk for the XE, very quick changing from compilier to editor and linker — quicker than for an IBM-PC. This will be released as a "FreeWare" product for a $35 donation when ready. Since most of the development work for the new ST's will use C, this is a good place to start.

The new 520ST's will come with LOGO as its prime language, so Ruth Ellsworth has again started to donate LOGO programs and articles for the Newsletter. She has also made up our first full LOGO disk for the library — see her article. Speaking of the libary, the **KoalaPad Utility Disk** (double-sided for $15) is really a great buy for anyone into graphics — one of our best disks. We also have a new public domain Modem disk (not the one we use and not from ACE) called **The Wizard** double sided for $10.

We have recently discovered that for some reason not known to anyone, our BBS, using the MPP modem, will not allow downloading from Atari 1030 Modem's. We will soon have a new, all machine language BBS which will run 300/1200 baud, etc, and be 10X or more faster than our present one (and hopefully with less bugs!).

I understand MPP (MicroBits) is now closed and undergoing re-organization. I was very sorry to hear that, since they were a great supporter of user groups and had very good products — wish them luck to get back again.

VP's RAMBLINGS

Soon the ST's will be upon us and I am just learning how to use the new XE. If Atari keeps on the way they are going with the new type machines I will be further behind then ever before. I don't know whether this is good or bad but it sure is fun. The new XE is a very good computer with a keyboard that feels good and with the added memory I hope to see new programs coming out for it real soon. I think that Atari is doing the right thing by not have too many computers in the same type of series. They really don't need the 65XE thus the 130XE compliments the 800 series. I hope that the ST series is the same way.

We have a new board up and running as we are beta testing it. It has all the features of the Forem board and more without all the problems. It is a password board so you have to log on as before but the higher levels really mean something and once you have your password to the board it really moves fast and gives you more time to do whatever you want on it. Anyway check it out and let us know what you think of it, also tell us of any features you would like to see.

We will be adding an ST section to the newsletter just as soon as we have programs and articles to print and as long as we have the material coming in we will print it and of course all the other features that we put in every month. Don't forget if you have an article or program that you want to share with the other members please send it to us and we will see that it is put in the newsletter.

— LARRY GOLD

THE 'C' DUFFER

2nd DIVOT

Last month we looked at some ways to get simple integers into and out of the new AceC programming system for Atari. An elementary C program demonstrated integer math operations.

AceC allows you to use Atari's floating point math. Most versions of C limit you to integer math. Floating point math insures fast-running programs. But integer math sure can slow down the programmer. I find floating point is cheap insurance against the kinds of errors which I make in fixed point math.

So, how do you use F.P. with AceC? None of the books on C can help you here. As usual, Atari is different. The 'bible' is a brief text file on your AceC disk, "FLOATING POINTS". If all else fails, read the instructions, right?

Our sample program will get you going. You must reserve six character bytes for each F.P. variable, and initialize any variable whose value is not set in the program (lines 50,60). Line 100 converts the input string to a F.P. number and puts it into a [6]. Then we 'fadd' a into sum and store the result in sum. After doing this for 12 iterations (line 70) we convert the index i to F.P., put it in n[6] and 'fdivide' sum by n to obtain the simple mean, which is stored in ave[6]. Lines 150 and 160 put the results on our screen with a label for each number. Line 170 holds the screen until you press another key.

The function fadd(a,b,c) is like C=A+B in BASIC, fdiv(a,b,c) is like C=A/B, and so on. You can also subtract, multiply, and raise a number to a power (which doesn't have to be an integer).

You now know how to enter numbers and characters into an AceC program and get output of some kind. That's about as much as I know, at the moment. It should be enough to let us learn to use C itself, from books like Kernighan and Ritchie's "The C Programming Language".

That's because the core of AceC is much like that of other C's, and I'm told that once you learn to use it, C is one of the very best programming languages. Once you get past the I/O barriers, that is.

So consider, as you listen to your disk drive earn its keep compiling yet another corrected version of a hot new C function, that most of the software for the new Atari ST computers is being written in C !

Finally, learning a new computer language which requires an editor, a compiler, and a linker — that's three passes through your trusty old Atari — before you see results (if any) can definitely give you a profound new appreciation for the all too familiar virtues of Atari's BASIC.

No matter how you play it, you win!

— Dick Barkley

DOLLAR A MEGABYTE

Page 18 of the latest **INFOWORLD** reports on the new Atari CD ROM shown at the June CES in Chicago. For $500 this peripheral may be added to your ST machine. This laser compact disk player uses standard compact disks which can contain 540 megabytes of data. The Atari CD player can also play your high-quality audio compact disks when you're not using the computer!

The display model in Chicago contained the entire 20-volume Grolier encyclopedia. Every one of the 9 million words in the encyclopedia is indexed. As a test, a search for the word "toothache" was made. It took 3 seconds to find every occurrence of the word in the encyclopedia. There is also a "browse" mode for leafing through the encyclopedia and a tree structure to access individual entries.

Lending libraries could make these large databases available to individual users who could not otherwise afford the large subscription costs and connect-time charges for the more expensive on-line databases. This development could really bring the information age into the average home!

Also shown at the June CES was an Atari 260 ST priced at $499. This version contains a built-in 1 megabyte 3.5" floppy and 256k RAM.

— Editors

LOGO LITTLE BITS

by Ruth Ellsworth

The new ST computers will soon be here. I find myself with the proverbial boxes of odds and ends (in this case disk boxes) as I get ready for the new machine. As part of the sorting out, I am giving the club disk of LOGO. The disk includes the LOGO listings I have included in the ACE Newsletter, the LOGO listings I wrote last year for TURTLE NEWS, two instant LOGOs (one for the younger set, and one for the even younger set), and odds and ends as time and space permit.

This month I want to mention the special primitives available in ATARI LOGO. Because of the way in which ATARI LOGO is accessed by the computer, users have little control over the computer memory or ability to change it. The special primitives in ATARI LOGO give very limited ability to affect computer memory. These primitives have the . (dot) at the beginning as a warning that they must be used carefully. They can destroy workspace. If workspace is destroyed through using them, one must restart LOGO and begin again. Four out of the five special primitives will not be used by beginners very often.

The special primitive used most is the .SETSCR command. This command adjusts the vertical and horizontal lines on the TV being used so that objects are in correct proportion. For example, squares can be made to look truely square rather than rectangle as some TV screens make them appear with the default .SETSCR value. For most TV screens .SETSCR .8 is the correct value.

The .PRIMITIVES command prints a list of all the Logo primitives. This is can be a handy reference guide during programming. We have also found it very useful when translating programs created at the children's school with different computers.

One of the nice things about ATARI LOGO is that machine language subroutines can be accessed through the .CALL command. This command allows ATARI LOGO to do things which LOGO was not designed to do. The .CALL command is followed by a number (address) representing the starting memory location for the subroutine.

Two special primitives allow LOGO users direct access to memory locations. These commands are .EXAMINE and .DEPOSIT. Both commands are followed by a number representing a memory location (address).

The .EXAMINE (number) command allows the user to read the contents of a memory location. It is used for such things as printer drivers which check the memory locations during the run of the program so that the printer will preform in the way desired.

The .DEPOSIT (number) command has very limited use in ATARI LOGO. The way LOGO is implemented makes most memory locations "off limits" to changes. There are, however a few locations which can be used to create fun or interesting effects. One of the things that is fun to do is to use the .DEPOSIT command to change the shape of the turtle. Page 154 of the Reference Manual gives the location and byte numbers that can be used. We have used those numbers with the collision registers to make the turtle look squashed.

Another fun thing we have done is to use the .DEPOSIT command to make titles for our programs that flip upside down and back. The FLIPPING TITLE modules at the end of this article can be used to introduce programs by adding the name of the program to be run after the line IF :VALUE × 10 [FLIPLET] in the TO COUNTER module.

The TO START module is required because the program uses a counter. Counters must be set in modules outside of the modules in which they are used, or the value of the counter will be reset to the original value each time. If the counter is reset to the original value the program will become a "forever" program and won't go anywhere.

A value of 2 placed in memory location (address) 755 makes text appear right side up on the screen. A value of 4 placed in that location makes text appear upside down. In the TO FLIPLET module we have first placed 2 in that location using the .DEPOSIT command. Be sure to remember when using this command that the address is placed directly after the command, followed by the value to be placed in that address.

SETCURSOR allows text to be placed anywhere on the screen. We have centered the text. It could be placed anywhere desired between 0 to 23 vertically, and 0 to 37 horizontally. However, horizontal location 37 is reserved. For all practical purposes horizontal numbering is from 0 to 36.

Also listed below is a PROCEDURE PRINTER. I have used it for the listings used in this article. We have found this to be one of the most useful of all LOGO programs. It makes it easy to keep track of each computer session. We especially like it to keep a record of programs that work the way we want them. I have made all the inputs variables so that one just loads PRINTOUT, calls it, then follows the prompts. The best thing about PRINTOUT is that one can print out only the procedures wanted, and in the order wanted. The reason for the repeated SETWRITE "P: and SETWRITE" commands is to keep the printer from echoing the screen. Echoing makes the inputs appear twice on the printed listing. In response to the prompt TYPE THE NAME OF THE PROCEDURE, type the names of all procedures wanted separated by spaces. Be sure to terminate all inputs with a RETURN.

BASIC BUGS

(Reprint: **FRANTIC, April/May, 1985)**

Have you ever tried to edit a BASIC program only to lose part or all of it due to a mysterious system lock-up? Here's a solution. Basic Fix transfers BASIC to an AUTORUN.SYS file and fixes this infamous bug, along with some other bugs in GET, minus zero, unary minus, and NOT.

The biggest change in this fixed BASIC is the precedence of NOT, which will be HIGH (equal to that of unary minus), instead of LOW. For example, with the original precedence, NOT 0+1 evaluates to 0, since 0+1 is executed before NOT. With the new precedence, this will evaluate to 2 because NOT 0 will be executed before +1. Proper use of parentheses will restore the original meaning.

Other visible changes include: PRINT -0 will give 0 instead of garbage, and −1 will evaluate to 1, as it should, instead of -1. See the Atari Basic Source Book, Appendix B, for further discussion of the bugs in Atari BASIC.

Another feature is the option to choose the screen color and margin defaults. Change the labeled DATA bytes to the combination you prefer.

This RAM-resident BASIC survives SYSTEM RESET and functions correctly in all GRAPHICS modes, at the cost of 4k less free memory. To return to this BASIC from DUP.SYS you **MUST** use option M and answer the prompt with address 9BBB. If MEM.SAV is applicable, it will be loaded and your program restored; otherwise the computer will return to BASIC with no program in memory.

To create your (almost) bug-free BASIC, boot up from a disk with DOS 2.0S, 68 free sectors, and no AUTORUN.SYS (or one you don't want), and RUN this program. then remove the BASIC cartridge, reboot, and you can SAFELY edit and run any Atari BASIC program. You can still use a utility such as MACRODOS by appending the BASIC AUTORUN.SYS file to it. An autoboot menu program can be used by appending it to the BASIC AUTORUN.SYS file. Use DOS menu function C with the /A option (see the DOS II Reference Manual for details).

— Trent Dudley

LOGO

```
TO START
MAKE "VALUE 1
CT
FLIPLET
END

TO FLIPLET
.DEPOSIT 755 2
SETCURSOR [15 12]
PR "TITLE
WAIT 60
.DEPOSIT 755 4
WAIT 60
.DEPOSIT 755 2
COUNTER
END

TO COUNTER
MAKE "VALUE :VALUE + 1
IF :VALUE < 10 [FLIPLET]
END
```

RAMTALKER
by Randy Holmes

```
1 REM ******************************
2 REM **   FROM S.T.A.T.U.S., VA. **
3 REM **   see Mar & Jun ACE for  **
4 REM **   articles and circuits  **
5 REM ** This version uses port 2 **
6 REM ** for XE and XL computers  **
7 REM **           by             **
8 REM **       Randy Holmes       **
9 REM ******************************
10 REM RAMTALKER VERSION 3.5 2/85
30 FOR I=0 TO 243:READ Z:POKE 1536+I,Z
:NEXT I:DIM PRINT$(167),FREQ$(1),AMP$(
1),TIM$(4),ARRAY(255)
35 FOR I=1 TO 167:READ A:LET PRINT$(I)
=CHR$(A):NEXT I
140 ON VAL(CHR$(ANS)) GOTO 160,200,240
,270,330,611
270 CLOSE #4:TRAP 270:POKE 752,1:? "GI
VE FILE NAME";:INPUT FN$:IF FN$="" THE
N 60
330 CLOSE #4:TRAP 330:POKE 752,1:? "GI
VE FILE NAME";:INPUT FN$:IF FN$="" THE
N 60
400 DATA 0,212,141,14,212,141,10,212,1
41,10,212,166,207,32,149,6,173,2,210,1
62
600 I=USR(ADR("hhh*LVd"),IO):CLOSE #IO
/16:RETURN :REM "*" AND "d" IN ADR STR
ING ARE INVERSE
611 REM SET UP GRAPH SCREEN
621 GRAPHICS 8:SETCOLOR 1,0,0:SETCOLOR
 2,0,12:COLOR 1
631 PLOT 90,10:DRAWTO 90,100:DRAWTO 26
5,100:PLOT 90,100:DRAWTO 40,150
641 DRAWTO 215,150:DRAWTO 265,100:DRAW
TO 265,10:DRAWTO 90,10:DRAWTO 40,60:DR
AWTO 40,150
651 FOR I=90 TO 265 STEP 5:PLOT I,99:P
LOT I,98:NEXT I:FOR I=10 TO 100 STEP 1
0:PLOT 91,I:PLOT 92,8:NEXT I:A=91
661 FOR I=100 TO 150 STEP 5:PLOT A,I:P
LOT A+1,I:A=A-5:NEXT I:GOSUB 900
671 REM SET UP AND CLEAR ARRAY USED
       FOR HOLDING SAMPLE VALUES
681 FOR I=0 TO 255:ARRAY(I)=0:NEXT I
684 REM WHICH PIECE OF SOUND (1.75 SEC
)
685 ? CHR$(125);"SECTION OF SOUND TO S
AMPLE?(1-4)":? " 1=0-1.75sec|2=1.75-3.
50sec":? " 3=3.50-4.25sec|4=4.25-7.0"
686 TRAP 60:INPUT SEC
687 IF SEC=1 THEN LOC=16384:GOTO 700
688 IF SEC=2 THEN LOC=20479:GOTO 700
689 IF SEC=3 THEN LOC=24574:GOTO 700
690 IF SEC=4 THEN LOC=28669:GOTO 700
691 GOTO 85
700 ? CHR$(125);"DURING PLOT, PRESS AN
Y KEY TO RETURN  TO MAIN MENU"
701 FOR Q=1 TO 35:Z=90+(Q*5):LOC=LOC+2
55:COUNT=0:OG=0:REM SET UP COUNT FOR 3
5 STEPS THROUGH MEMORY
711 FOR I=0 TO 255:ARRAY(I)=0:NEXT I:R
EM SET UP TO COUNT ALL VALUES AT LOC
721 FOR I=1 TO 225:REM 225 SAMPLES=1/2
0 OF A SECOND
731 SMPL=PEEK(I+LOC):REM GET SAMPLE VA
L
741 ARRAY(SMPL)=ARRAY(SMPL)+1:REM SET
COUNTER ARRAY
751 NEXT I
761 REM PLOT THE INFO ON SCREEN
771 TRAP 811:FOR W=1 TO 50 STEP 0.196:
COUNT=COUNT+1:SMPL=ARRAY(COUNT):IF COU
NT)255 THEN COUNT=0
781 IF OG=1 THEN 801
791 PLOT (Z-W),(100+W-SMPL):OG=1
801 DRAWTO (Z-W),(100+W-SMPL):OG=1
805 IF PEEK(764)<>255 THEN GOTO 60
811 NEXT W
821 NEXT Q
830 ? CHR$(125);"PRESS ANY KEY TO RETU
RN TO MAIN MENU"
840 IF PEEK(764)=255 THEN 840
850 GOTO 60
900 REM PUT LABELS ON SCREEN
950 TIM$="TIME":RESTORE 2070
960 MLP=ADR(PRINT$)
1100 RESTORE 2070:FOR I=0 TO 8:READ FR
EQ$:A=USR(MLP,195,2+I,ADR(FREQ$),LEN(F
REQ$)):NEXT I
1110 A=USR(MLP,98,0,ADR(TIM$),LEN(TIM$
))
1130 FOR I=0 TO 8:READ AMP$:A=USR(MLP,
198-I,10+I,ADR(AMP$),LEN(AMP$)):NEXT I
:RETURN
2000 DATA 104,201,4,240,9,170,240,5,10
4,104,202,208,251,96,104,133,215,104,1
33,214,104,104,168,104,133,217,104
2010 DATA 133,216,104,104,240,136,133,
212,24,165,214,101,88,133,214,165,89,1
01,215,133,215,152,240,15,165,214
2020 DATA 105,64,133,214,165,215,105,1
,133,215,136,208,241,132,221,160,0,132
,220,177,216,160,0,170,16,1,136
2030 DATA 132,213,138,41,96,208,4,169,
64,16,14,201,32,208,4,169,0,16,6,201,6
4,208,2,169,32,133,218,138,41,31
2040 DATA 5,218,133,218,169,0,162,8,6,
218,42,202,208,250,109,244,2,133,219,1
64,221,177,218,69,213,164,220,145
2050 DATA 214,200,132,220,196,212,208,
182,24,165,214,105,40,133,214,144,2,23
0,215,230,221,169,8,197,221,208,159
2060 DATA 96,207,F,R,E,Q,U,E,N,C,Y,E,D
,U,T,I,L,P,M,A
2070 DATA F,R,E,Q,U,E,N,C,Y,E,D,U,T,I,
L,P,M,A
```

ACTION

```
PROC UPCASE(CARD A2,A1)
;
; RETURNS THE STRING S1 THROUGH S2
; CONVERTED TO UPPERCASE
;
; A1,A2 ARE THE ADDRESSES OF S1,S2
; RESPECTIVELY
;
BYTE ARRAY S1,S2
BYTE I
 S1=A1
 S2=A2
 S2(0)=S1(0)
 FOR I=1 TO S1(0)
 DO
  IF 'z)=S1(I) AND S1(I)='a THEN
   S2(I)=S1(I)+('A-'a)
  ELSE
   S2(I)=S1(I)
  FI
 OD
RETURN

PROC LOWCASE(CARD A2,A1)
;
; RETURNS THE STRING S1 THROUGH S2
; CONVERTED TO LOWERCASE
;
; A1,A2 ARE THE ADDRESSES OF S1,S2
; RESPECTIVELY
;
BYTE ARRAY S1,S2
BYTE I
 S1=A1
 S2=A2
 S2(0)=S1(0)
 FOR I=1 TO S1(0)
 DO
  IF 'Z)=S1(I) AND S1(I)='A THEN
   S2(I)=S1(I)-('A-'a)
  ELSE
   S2(I)=S1(I)
  FI
 OD
RETURN
```

SOLITAIRE by John Kelly

```
   RAPHICS 18:POKE 756,209
2 REM **************************
3 REM ** SOLITAIRE by John Kelly  **
4 REM **                         **
5 REM **      ACE Newsletter     **
6 REM **     3662 Vine Maple     **
7 REM **    Eugene, OR 97405     **
8 REM **         $14 year        **
9 REM **************************
10 POSITION 5,3:? #6;"SOLITAIRE":POSIT
ION 3,5:? #6;"TRANSLATED BY":POSITION
3,7:? #6;"John P Kelley"
15 POSITION 4,9:? #6;"VERSION TWO":GOT
O 1110
20 COLOR 1:FOR AZ=Y TO Y+30:PLOT X,AZ:
DRAWTO X+30,AZ:NEXT AZ
22 C=VA:ON SU GOTO 36,46,24,28
23 REM DIAMONDS
24 COLOR 0:FOR ZA=0 TO 9:FOR AZ=1 TO Z
A STEP 2:PLOT X+15+AZ,Y+ZA+6:PLOT X+15
-AZ,Y+ZA+6:NEXT AZ:NEXT ZA
25 FOR ZA=9 TO 0 STEP -1:FOR AZ=1 TO Z
A STEP 2:PLOT X+15+AZ,Y-ZA+25:PLOT X+1
5-AZ,Y-ZA+25:NEXT AZ:NEXT ZA:GOTO 56
27 REM HEARTS
28 COLOR 0:FOR ZA=0 TO 10:FOR AZ=1 TO
ZA STEP 2:PLOT X+15+AZ,Y-ZA+25:PLOT X+
15-AZ,Y-ZA+25:NEXT AZ:NEXT ZA
30 FOR ZA=11 TO 15:FOR AZ=1 TO 10 STEP
 2:PLOT X+15+AZ,Y-ZA+25:PLOT X+15-AZ,Y
   +25:NEXT AZ:NEXT ZA
   Z=-2:FOR ZA=16 TO 18:Z=Z+2:FOR AZ=Z
 TO 8-Z STEP 2:PLOT X+15+AZ,Y-ZA+25:PL
OT X+15-AZ,Y-ZA+25:NEXT AZ:NEXT ZA
34 GOTO 56
35 REM SPADES
36 COLOR 0:FOR AZ=0 TO 9:PLOT X+15+AZ,
Y+AZ+6:DRAWTO X+15-AZ,Y+AZ+6:NEXT AZ
38 FOR AZ=10 TO 12:PLOT X+15+9,Y+AZ+6:
DRAWTO X+15-9,Y+AZ+6:NEXT AZ
40 Z=-2:FOR ZA=13 TO 15:Z=Z+2:FOR AZ=Z
 TO 8-Z:PLOT X+15+AZ,Y+ZA+6:PLOT X+15-
AZ,Y+ZA+6:NEXT AZ:NEXT ZA
42 Z=0:FOR AZ=13 TO 19:Z=Z+1:PLOT X+15
+Z,Y+AZ+6:DRAWTO X+15-Z,Y+AZ+6:NEXT AZ

44 GOTO 56
45 REM CLUBS
46 COLOR 0:AZ=X+11:ZA=Y+6:GOSUB 52:AZ=
X+5:ZA=Y+13:GOSUB 52:AZ=X+17:GOSUB 52
48 Z=0:FOR AZ=14 TO 21:Z=Z+0.65:PLOT X
+15+Z,Y+AZ+5:DRAWTO X+15-Z,Y+AZ+5:NEXT
 AZ
50 FOR Z=X+14 TO X+16:PLOT Z,Y+13:DRAW
TO Z,Y+19:NEXT Z:GOTO 56
52 PLOT AZ+2,ZA:DRAWTO AZ+6,ZA:PLOT AZ
+2,ZA+7:DRAWTO AZ+6,ZA+7:PLOT AZ+1,ZA+
1:DRAWTO AZ+7,ZA+1
54 PLOT AZ+1,ZA+6:DRAWTO AZ+7,ZA+6:FOR
 Z=ZA+2 TO ZA+5:PLOT AZ,Z:DRAWTO AZ+8,
Z:NEXT Z:RETURN
56 CHR=ASC(C$(C,C)):IF CHR<96 THEN CHR
=CHR-32*(CHR)31)+64*(CHR<32)
58 CPOS=CHR*8+(PEEK(756)*256)
60 FOR AZ=0 TO 7:POKE (AZ*40)+MEM+(X/8
)+(Y*40),255-PEEK(CPOS+AZ):NEXT AZ:RET
URN
70 SU=INT(NUM/100)
80 VA=NUM-100*SU
90 RETURN
100 IF HF THEN GOSUB 1480:RETURN
105 IF IN)48 AND IN(7)<4 THEN 1800
110 IF IN)49 THEN 1650
120 FOR K=0 TO 2:OD(IN(7)+K)=D(IN+K):N
EXT K:IN=IN+3:IN(7)=IN(7)+2
125 X=X(7):Y=Y(13):NUM=OD(IN(7)):GOSUB
 70:GOSUB 20:IN(7)=IN(7)+1
140 RETURN
150 POKE 656,0:IF OC<7 THEN POKE 657,4
*OC+4:? " "
160 IF OC=7 THEN POKE 657,35:? " "
170 POKE 656,0:IF CU<7 THEN POKE 657,4
*CU+4:? "A"
180 IF CU=7 THEN POKE 657,35:? "A"
190 RETURN
200 CU=CU+1:IF CU)7 THEN CU=7
210 GOSUB 150
220 OC=CU
230 RETURN
240 CU=CU-1:IF CU<0 THEN CU=0
250 GOSUB 150
260 OC=CU
270 RETURN
280 IF HF THEN GOSUB 1480:RETURN
290 ST=CU
300 IF IN(CU)=0 THEN GOSUB 1510:RETURN

310 IF CU=7 THEN NUM=OD(IN(7)-1):GOTO
330
320 NUM=C(CU,0)
330 HF=1
340 J=0:IF CU=7 THEN COLOR 1:FOR I=0 T
O 30:PLOT X(CU)+I,Y(13):DRAWTO X(CU)+I
,Y(13)+30:GOTO 370
350 IF P(CU,0)=0 THEN COLOR 0:FOR I=0
TO 30:PLOT X(CU)+I,Y(J):DRAWTO X(CU)+I
,Y(J)+30:GOTO 370
360 FOR I=0 TO 30 STEP 2:COLOR 0:PLOT
X(CU)+I,Y(J):DRAWTO X(CU)+I,Y(J)+30
365 IF I<30 THEN COLOR 1:PLOT X(CU)+I+
1,Y(J):DRAWTO X(CU)+I+1,Y(J)+30
370 NEXT I
380 IF CU=7 THEN GOSUB 150:RETURN
390 J=12*(IN(CU)-1)+32
395 COLOR 0:FOR I=0 TO 30:PLOT X(CU)+I
,Y(0)+31:DRAWTO X(CU)+I,Y(0)+J:NEXT I
400 RETURN
410 IF  NOT HF THEN GOSUB 1520:RETURN

420 IF CU=7 THEN GOSUB 590:RETURN
430 IF ST=CU THEN GOSUB 750:RETURN
440 IF IN(CU)=0 THEN GOSUB 630:RETURN

450 NUM=C(CU,IN(CU)-1)
460 GOSUB 70:TS=SU:TV=VA
470 IF ST=7 THEN NUM=OD(IN(7)-1):GOTO
490
480 NUM=C(ST,0)
490 GOSUB 70:IF ((TS=1) OR (TS=2)) AND
 ((SU=1) OR (SU=2)) THEN GOSUB 1530:RE
TURN
500 IF ((TS=3) OR (TS=4)) AND ((SU=3)
OR (SU=4)) THEN GOSUB 1540:RETURN
510 IF TV()VA+1 THEN GOSUB 1550:RETURN

520 IF ST=7 THEN GOSUB 700:RETURN
530 FOR I=0 TO IN(ST)-1:NUM=C(ST,I):C(
CU,IN(CU))=NUM:GOSUB 70:X=X(CU):Y=Y(IN
(CU)):GOSUB 20:IN(CU)=IN(CU)+1
540 C(ST,I)=0:NEXT I:IN(ST)=0:HF=0
550 IF P(ST,0)=0 THEN RETURN
560 NUM=P(ST,0):GOSUB 70:X=X(ST):Y=Y(0
):GOSUB 20:C(ST,IN(ST))=NUM:IN(ST)=1
570 FOR I=0 TO 4:P(ST,I)=P(ST,I+1):NEX
T I:P(ST,5)=0
580 RETURN
590 IF ST()7 THEN GOSUB 1560:RETURN
600 NUM=OD(IN(7)-1):GOSUB 70:X=X(CU):Y
=Y(13):GOSUB 20:GOSUB 150
610 HF=0
620 RETURN
630 IF ST=7 THEN NUM=OD(IN(7)-1):GOTO
650
640 NUM=C(ST,0)
650 GOSUB 70
660 IF VA()13 THEN GOSUB 1570:RETURN
670 IF ST=7 THEN GOSUB 700:RETURN
680 GOSUB 530
690 RETURN
700 X=X(CU):Y=Y(IN(CU)):C(CU,IN(CU))=N
UM:GOSUB 20:IN(CU)=IN(CU)+1
710 IN(7)=IN(7)-1:OD(IN(7))=0:HF=0
720 IF IN(7)=0 THEN COLOR 0:X=X(7):Y=Y
(13):FOR I=0 TO 30:PLOT X+I,Y:DRAWTO X
+I,Y+30:NEXT I:RETURN
730 NUM=OD(IN(7)-1):GOSUB 70:X=X(7):Y=
Y(13):GOSUB 20
740 RETURN
750 FOR I=0 TO IN(CU)-1:NUM=C(CU,I):GO
SUB 70:X=X(CU):Y=Y(I):GOSUB 20:NEXT I
760 HF=0
```

solitaire cont

```
770 RETURN
780 NUM=OD(IN(7)-1):GOSUB 70:FL=1
785 IF (F(SU)<>VA-1) AND (F(SU)=0) THE
N GOSUB 1580:RETURN
790 IF F(SU)<>VA-1 THEN TV=F(SU):GOSUB
 1550:RETURN
800 GOSUB 980
810 OD(IN(CU))=0
820 IF IN(CU)=0 THEN GOSUB 720:RETURN

830 GOSUB 730
835 GOSUB 150
840 RETURN
850 IF P(CU,0)=0 THEN COLOR 0:FOR I=0
TO 30:PLOT X(CU)+I,Y(0):DRAWTO X(CU)+I
,Y(0)+30:GOTO 875
860 FOR I=0 TO 30 STEP 2:COLOR 0:PLOT
X(CU)+I,Y(0):DRAWTO X(CU)+I,Y(0)+30
870 IF I<30 THEN COLOR 1:PLOT X(CU)+I+
1,Y(0):DRAWTO X(CU)+I+1,Y(0)+30
875 NEXT I:C(CU,0)=P(CU,0)
880 IF P(CU,0)=0 THEN RETURN
890 NUM=C(CU,0):X=X(CU):Y=Y(0):GOSUB 7
0:GOSUB 20
900 IN(CU)=1
910 FOR I=0 TO 4:P(CU,I)=P(CU,I+1):NEX
T I:P(CU,5)=0
920 RETURN
930 IF HF THEN RETURN
935 FL=0
940 IF IN(CU)=0 THEN GOSUB 1510:RETURN

950 IF CU=7 THEN GOSUB 780:RETURN
960 NUM=C(CU,IN(CU)-1):GOSUB 70
965 IF (F(SU)<>VA-1) AND (F(SU)=0) THE
N GOSUB 1580:RETURN
970 IF F(SU)<>VA-1 THEN TV=F(SU):GOSUB
 1550:RETURN
980 X=X(7)
990 IF SU=1 THEN Y=Y1
1000 IF SU=2 THEN Y=Y2
1010 IF SU=3 THEN Y=Y3
1020 IF SU=4 THEN Y=Y4
1030 GOSUB 20:F(SU)=VA
1040 IN(CU)=IN(CU)-1:IF FL THEN RETURN

1050 C(CU,IN(CU))=0
1060 IF IN(CU)=0 THEN GOSUB 850:RETURN

1070 X=X(CU):Y=Y(IN(CU)-1):NUM=C(CU,IN
(CU)-1):GOSUB 70:GOSUB 20
1080 COLOR 0
1090 FOR I=31 TO 45:PLOT X(CU),Y(IN(CU
)-1)+I:DRAWTO X(CU)+30,Y(IN(CU)-1)+I:N
EXT I
1100 RETURN
1110 CLR :DIM TX$(91),AZ$(7),C$(13),C(
6,11),P(6,5),D(51),OD(23),F(4),X(7),Y(
13),IN(7):OPEN #1,4,0,"K"
1115 TX$(91)="#":TX$(1,1)=" ":TX$(2)=T
X$(1):C$="A23456789TJQK":AZ$=TX$
1120 FOR I=1 TO 13:AZ$="       ":READ
AZ$:TX$(I*7-6,I*7)=AZ$:NEXT I
1150 FOR I=0 TO 6:FOR J=0 TO 5:C(I,J)=
0:P(I,J)=0:NEXT J:FOR J=6 TO 11:C(I,J)
=0:NEXT J:NEXT I
1160 FOR I=0 TO 23:OD(I)=0:NEXT I
1170 FOR I=0 TO 4:F(I)=0:NEXT I
1180 FOR I=0 TO 7:X(I)=21+I*32:Y(I)=I*
12:NEXT I:X(7)=269
1190 FOR I=8 TO 12:Y(I)=I*12:NEXT I
1200 Y1=0:Y2=32:Y3=64:Y4=96:Y(13)=128
1210 POKE 756,224
1220 IN=0:FOR I=1 TO 4:FOR J=1 TO 13:D
(IN)=100*I+J:IN=IN+1:NEXT J:NEXT I
1230 FOR I=51 TO 0 STEP -1:X=INT(RND(0
)*(I+1)):T=D(X):D(X)=D(I):D(I)=T:NEXT
I
1240 IN=0:FOR I=1 TO 6:FOR J=0 TO I-1:
P(I,J)=D(IN):IN=IN+1:NEXT J:NEXT I
1250 FOR I=0 TO 6:C(I,0)=D(IN):IN=IN+1
:NEXT I
1260 GRAPHICS 8:POKE 710,0:POKE 752,1:
COLOR 1:POKE 709,15:POKE 712,4:MEM=PEE
K(88)+PEEK(89)*256
1280 FOR X=254 TO 258 STEP 2:PLOT X,0:
DRAWTO X,160:NEXT X
1290 FOR I=0 TO 6:NUM=C(I,0):GOSUB 70:
X=X(I):Y=Y(0):GOSUB 20:NEXT I
1300 FOR I=0 TO 6:IN(I)=1:NEXT I:IN(7)
=0
1310 GOSUB 100:GOSUB 1460
1320 CU=0:OC=0:X=X(CU):GOSUB 150
1330 POKE 764,255:GET #1,A:A=A-32*(A)9
0)
1340 IF A=42 THEN GOSUB 200:GOTO 1330
1350 IF A=43 THEN GOSUB 240:GOTO 1330
1360 IF A=78 THEN GOSUB 100:GOTO 1330
1370 IF A=80 THEN GOSUB 280:GOTO 1330
1380 IF A=68 THEN GOSUB 410:GOTO 1330
1390 IF A=70 THEN GOSUB 930:GOTO 1620
1400 IF A=69 THEN 1420
1410 GOTO 1330
1420 GOSUB 1600:? "DO YOU WANT TO END
THE GAME? (Y/N)";:GET #1,A
1430 IF A<>89 THEN GOSUB 1460:GOTO 133
0
1440 RUN
1450 GOSUB 1600:END
1460 GOSUB 1600:? "ARROWS MOVE, E=END
GAME F=FOUNDATION  N=NEXT CARD, P=PICK
 UP CARDS, D=DROP";
1470 RETURN
1480 GOSUB 1600:? "YOU'VE ALREADY PICK
ED UP A CARD";:GOTO 1610
1490 GOSUB 1600:? "THERE ARE NO MORE
ARDS IN THE DECK   YOU MUST PLAY WI
THE CARDS SHOWING";:GOTO 1610
1510 GOSUB 1600:? "THERE ARE NO CARDS
HERE TO PICK UP";:GOTO 1610
1520 GOSUB 1600:? "YOU DO NOT HAVE ANY
 CARDS TO DROP";:GOTO 1610
1530 GOSUB 1600:? "YOU CAN'T PLAY BLAC
K ON BLACK";:GOTO 1610
1540 GOSUB 1600:? "YOU CAN'T PLAY RED
ON RED";:GOTO 1610
1550 GOSUB 1600:? "YOU CAN'T DROP A";T
X$(VA*7-6,VA*7):? "ON TOP OF A";TX$(TV
*7-6,TV*7);:GOTO 1610
1560 GOSUB 1600:? "YOU CAN'T DROP CARD
S HERE";:GOTO 1610
1570 GOSUB 1600:? "YOU CAN ONLY DROP A
 KING HERE";:GOTO 1610
1580 GOSUB 1600:? "START YOUR FOUNDATI
ON WITH AN ACE";:GOTO 1610
1590 DATA N ACE, TWO, THREE, FOUR, FIV
E, SIX, SEVEN,N EIGHT, NINE, TEN, JACK
, QUEEN, KING
1600 POKE 656,2:POKE 657,2:? CHR$(156)
;CHR$(156);CHR$(253);:RETURN
1610 FOR PAUSE=1 TO 300:NEXT PAUSE:GOS
UB 1460:RETURN
1620 IF F(1)<13 OR F(2)<13 OR F(3)<13
OR F(4)<13 THEN GOTO 1330
1630 GOSUB 1600:? "YOU MIGHT CARE TO
 PLAY AGAIN? (Y/N)";:GET #1,A:IF A<>78
 THEN RUN
1635 IF A=78 THEN RUN "D:MENUPLUS"
1640 GRAPHICS 0:CLR
1650 IN(7)=IN(7)-1:AVER=0:BVER=0:IF ON
=15 THEN BVER=D(51):AVER=D(50)
1660 IF IN=51 THEN BVER=D(51)
1670 FOR VV=0 TO IN(7):D(51-VV)=OD(IN(
7)-VV):NEXT VV
1680 IF BVER>0 THEN D(51-VV)=BVER
1690 IF AVER>0 THEN VV=VV+1:D(51-VV)=A
VER
1700 IN=51-VV:IN(7)=0:IN=IN+1:IF BVER>
0 THEN IN=IN-1:GOTO 120
1800 IF IN>51 THEN 1490
1810 OD(IN(7))=D(IN):IN=IN+1:GOTO 125
```

LEXICON

```
   M ******************************
  .M ** SEE JUNE '85 ACE pg. 3   **
3 REM **    LEXICON  JULY 1985    **
4 REM **      John R. Kelley      **
5 REM **       ACE NEWSLETTER     **
6 REM **       3662 VineMaple     **
7 REM **      Eugene, Or 97405    **
8 REM **          $14 year        **
9 REM ******************************
10 GRAPHICS 0:POKE 752,1
15 POKE 710,0
20 DL=PEEK(560)+256*PEEK(561)
25 POKE DL+7,7:POKE DL+8,6:POSITION 4,
2:? "L E X I C O N "
30 POSITION 21,2:? "synonyms - antonym
s"
35 POSITION 2,4:? " This program provi
des the user two   hundred words, thei
r SYNONYMS and "
40 ? "ANTONYMS to develop word skills.
 The  WORDS are commonly used ones and
 are"
45 ? "presented in a multiple choice f
ormat to make it easier to choose the
cor-"
50 ? "rect answer. To use the program,
 sel- ect SYNONYM or ANTONYM and the n
umber"
55 ? "of WORDS to be learned. The sour
ce of the WORDS was WEBSTERS New Colle
g iate"
     "Dictionary.
                HAPPY LEXICOGRAPHY "
80 POKE DL+23,6:POKE DL+24,6:POSITION
5,17:? "DEVISED BY":POSITION 23,17:? "
JOHN R. KELLEY"
85 POSITION 7,19:? "Hit START to beg
in words."
90 IF PEEK(53279)=6 THEN GRAPHICS 0:GO
TO 100
95 GOTO 90
100 DIM WORD$(20),SYN$(20),ANT$(20),AN
S$(20),TEMP$(20),NUM$(3),BL$(20),T(3)
102 DIM SAME$(20),DIFF$(20),L$(3)
105 SAME$="PICK THE SYNONYM"
107 DIFF$="PICK THE ANTONYM"
110 REM
120 TEMP$=ANS$:BL$=ANS$:B=307
125 OPEN #1,4,0,"K:"
130 GRAPHICS 0:POKE 709,4:POKE 710,15:
POKE 712,15:POKE 752,1
132 POKE 708,66:SCORE=0
135 ? :? "To pick the SYNONYM, type 'S
Y'";:? :? "and hit RETURN"
137 ? :?
140 ? :? "To pick the ANTONYM, type 'A
N'";:? :? "and hit RETURN"
142 INPUT L$
143 IF L$="SY" THEN 150
145 IF L$="AN" THEN 1330
147 IF L$()"SY" OR L$()"AN" THEN 142
150 ? :? :? "How many words would you
like";:INPUT NUMBER
155 FOR TOTAL=1 TO NUMBER:? "%"
157 POKE DL+7,7:POKE DL+8,6:POKE 708,2
12
160 POSITION 1,2:? SAME$
161 POSITION 25,2:? "WORD #";TOTAL
162 POKE DL+9,7:POKE DL+10,7
165 POSITION 21,3
170 ANS=1995+INT(200*RND(0)+1)*5:RESTO
RE ANS:READ WORD$:READ ANS$:READ ANT$
175 ? WORD$:? :? :?
180 FOR I=0 TO 3
190 T(I)=1995+INT(200*RND(0)+1)*5:IF T
(I)=ANS THEN 190
200 FOR J=0 TO 3:IF I()J AND T(I)=T(J)
 THEN POP :GOTO 190
210 NEXT J:NEXT I:J=1+INT(RND(0)*5):A=
0:FOR I=1 TO 5:? I;") ";
220 IF I()J THEN RESTORE T(A):READ WOR
D$:READ TEMP$:READ ANT$:? TEMP$:A=A+1:
GOTO 250
230 ? ANS$
250 NEXT I:?
255 N=N1
260 GOSUB 800:IF N1(1 OR N1)5 THEN POS
ITION C,R:GOTO 260
270 IF N1()J THEN 310
280 SCORE=SCORE+1:GOSUB 1200:? :POKE 8
5,20:? "Good!":POKE 20,0
290 IF PEEK(20)(90 THEN 290
300 GOTO 340
310 GOSUB 1300:POSITION 0,2:? "Sorry,
the answer: "
315 POKE DL+7,7:POKE DL+8,7
320 POSITION 25,2:? ANS$:? :? "Hit RET
URN to continue."
330 POKE 764,255:GET #1,A:IF A()155 TH
EN 330
340 ? "%":NEXT TOTAL
400 POSITION 1,2:? SAME$
410 POSITION 2,5:? "Out of ";NUMBER;"
questions, you answered":?
420 ? SCORE;" correctly. Your score is
: ";
430 ? INT((SCORE/NUMBER)*100+0.5);"%,"

435 ? :? "RETURN for more words, * t
o quit."
440 POKE 764,255:GET #1,A:IF A()155 AN
D A()42 THEN 440
450 IF A=155 THEN 130
455 ? :? "                    "
460 ? "      SEE YOU LATER!! "
470 ? "                    ":FOR I=1
 TO 333:NEXT I:GRAPHICS 0:END
800 C=PEEK(85):R=PEEK(84)
805 POSITION C,R:? BL$:TRAP 805:Y=1:S=
0:NUM$=""
810 POSITION C+Y-1,R:? "@";:POKE 764,2
55:GET #1,A:POSITION C+Y-1,R:IF A()126
 THEN 825
815 IF Y)1 THEN Y=Y-1:? " ";:NUM$=NUM$
(1,Y)
820 GOTO 810
825 IF A=155 THEN ? " ":GOTO 840
830 NUM$(Y,Y)=CHR$(A):IF Y=1 THEN S=A
835 ? CHR$(A);:Y=Y+1:GOTO 810
840 POKE ADR(NUM$),S:N1=VAL(NUM$):RETU
RN
1200 REM *** CORRECT ANSWER ***
1210 SOUND 0,35,10,10:FOR DELAY=1 TO 1
5:NEXT DELAY
1220 SOUND 0,31,10,10:FOR DELAY=1 TO 1
5:NEXT DELAY
1230 SOUND 0,29,10,10:FOR DELAY=1 TO 1
5:NEXT DELAY
1240 SOUND 0,0,0,0
1250 RETURN
1300 REM *** WRONG ANSWER S/R ***
1310 SOUND 0,70,12,8:FOR J=0 TO 6:FOR
K=0 TO 7:NEXT K:NEXT J:SOUND 0,0,0,0:?
 "%":RETURN
1330 REM
1340 ? :? :? "How many words would you
 like";:INPUT NUMBER
1345 FOR TOTAL=1 TO NUMBER:? "%"
1350 POKE DL+7,7:POKE DL+8,6:POKE 708,
212
1355 POSITION 1,2:? DIFF$
1360 POSITION 25,2:? "WORD #";TOTAL
1365 POKE DL+9,7:POKE DL+10,7
1370 POSITION 21,3
1375 ANS=1995+INT(200*RND(0)+1)*5:REST
ORE ANS:READ WORD$:READ SYN$:READ ANS$

1380 ? WORD$:? :? :?
1385 FOR I=0 TO 3
1390 T(I)=1995+INT(200*RND(0)+1)*5:IF
T(I)=ANS THEN 1390
1395 FOR J=0 TO 3:IF I()J AND T(I)=T(J
) THEN POP :GOTO 1390
1400 NEXT J:NEXT I:J=1+INT(RND(0)*5):A
=0:FOR I=1 TO 5:? I;") ";
1410 IF I()J THEN RESTORE T(A):READ WO
RD$:READ SYN$:READ TEMP$:? TEMP$:A=A+1
:GOTO 1430
1420 ? ANS$
1430 NEXT I:?
1435 N=N1
```

by John Kelly

```
1440 GOSUB 800:IF N1<1 OR N1>5 THEN PO
SITION C,R:GOTO 1440
1445 IF N1<>J THEN 1465
1450 SCORE=SCORE+1:GOSUB 1200:? :POKE
85,20:? "Good!":POKE 20,0
1455 IF PEEK(20)<90 THEN 1455
1460 GOTO 1480
1465 GOSUB 1300:POSITION 0,2:? "sorry,
 the answer: "
1467 POKE DL+7,7:POKE DL+8,7
1470 POSITION 25,2:? ANS$:? :? "Hit [R
ETURN] to continue."
1475 POKE 764,255:GET #1,A:IF A<>155 T
HEN 1475
1480 ? "K":NEXT TOTAL
1490 POSITION 1,2:? DIFF$
1495 POSITION 2,5:? "Out of ";NUMBER;"
 questions, you answered":?
1500 ? SCORE;" correctly. Your score i
s: ";
1505 ? INT((SCORE/NUMBER)*100+0.5);"%.
"
1510 ? :? "[RETURN] for more words, [*
] to quit."
1515 POKE 764,255:GET #1,A:IF A<>155 A
ND A<>42 THEN 1515
1520 IF A=155 THEN 130
1525 ? :? "[                    ]"
1530 ? "[    SEE YOU LATER    ]"
1535 ? "[                    ]":FOR I=
1 TO 333:NEXT I:GRAPHICS 0:END
2000 DATA ABATE,DIMINISH,INCREASE
2005 DATA ABILITY,TALENT,INCAPACITY
2010 DATA ABSENT,AWAY,PRESENT
2015 DATA ACCOST,ADDRESS,AVOID
2020 DATA ADVERSE,CONTRARY,FAVORABLE
2025 DATA AGITATE,AROUSE,PLACATE
2030 DATA ALLOW,PERMIT,FORBID
2035 DATA ANXIETY,FEAR,ASSURANCE
2040 DATA APPEASE,PACIFY,AROUSE
2045 DATA ARRANGE,DEVISE,CONFUSE
2050 DATA ASSIST,HELP,HINDER
2055 DATA ATTACH,AFFIX,SEPARATE
2060 DATA AUTHENTIC,GENUINE,FALSE
2065 DATA AWKWARD,GAUCHE,ADROIT
2070 DATA BANISH,DISMISS,ACCEPT
2075 DATA BANAL,TRITE,NOVEL
2080 DATA BEAUTIFUL,COMELY,HOMELY
2085 DATA BEGET,PRODUCE,DESTROY
2090 DATA BENEATH,BELOW,ABOVE
2095 DATA BIGOT,FANATIC,LIBERAL
2100 DATA BLESS,EXTOL,CURSE
2105 DATA BONDAGE,SLAVERY,FREEDOM
2110 DATA BORDER,EDGE,CENTER
2115 DATA BUOYANT,RESILIENT,DEJECTED
2120 DATA BURDEN,ENCUMBER,LIGHTEN
2125 DATA CALAMITY,ADVERSITY,FORTUNE
2130 DATA CANCEL,ERASE,ENACT
2135 DATA CALLOUS,UNFEELING,SENSITIVE
2140 DATA CAUSE,INCENTIVE,EFFECT
2145 DATA CAUTION,PRUDENCE,ABANDON
2150 DATA CHARGE,ACCUSE,ACQUIT
2155 DATA CHERISH,VALUE,REJECT
2160 DATA CLEVER,TALENTED,CLUMSY
2165 DATA COMPOSE,FASHION,DESTROY
2170 DATA CONCISE,INCISIVE,VERBOSE
2175 DATA CONTAIN,EMBODY,EMIT
2180 DATA COURAGE,BRAVERY,COWARDICE
2185 DATA CRAFTY,CUNNING,CANDID
2190 DATA CURRENT,PRESENT,ANCIENT
2195 DATA CURT,ABRUPT,SMOOTH
2200 DATA DAMAGE,DEFACE,REPAIR
2205 DATA DEBASE,DEGRADE,RESTORE
2210 DATA DECEIT,GUILE,CANDOR
2215 DATA DEFEAT,CONQUER,SURRENDER
2220 DATA DELAY,DETAIN,HASTEN
2225 DATA DEMUR,DOUBT,DECIDE
2230 DATA DEPART,LEAVE,REMAIN
2235 DATA DETACH,CURTAIL,ENLARGE
2240 DATA DISCREET,TACTFUL,RUDE
2245 DATA DIVERSE,DISTINCT,ALIKE
2250 DATA DOGMA,DOCTRINE,CONDUCT
2255 DATA DOUBT,DISTRUST,BELIEF
2260 DATA DRIVE,FORCE,INDUCE
2265 DATA EAGER,FERVENT,APATHETIC
2270 DATA EASY,SIMPLE,DIFFICULT
2275 DATA EDICT,DECREE,INTENTION
2280 DATA ELECT,CHOOSE,REJECT
2285 DATA ELEVATE,UPLIFT,DEPRESS
2290 DATA ELUDE,AVOID,MEET
2295 DATA EMERGE,APPEAR,VANISH
2300 DATA EMPLOY,UTILIZE,DISCARD
2305 DATA ENDURE,SUSTAIN,SUCCUMB
2310 DATA ENTIRE,TOTAL,PARTIAL
2315 DATA ERASE,CANCEL,ENACT
2320 DATA EXALT,ELEVATE,DEBASE
2325 DATA FACE,FRONT,BACK
2330 DATA FAILURE,FIASCO,SUCCESS
2335 DATA FEAR,DREAD,COURAGE
2340 DATA FICKLE,CHANGEABLE,CONSTANT
2345 DATA FOLLOW,COMPLY,AVOID
2350 DATA FORM,FASHION,DESTROY
2355 DATA FORWARD,ADVANCE,RETARD
2360 DATA FREEDOM,LIBERTY,BONDAGE
2365 DATA FRIGHTEN,SCARE,SOOTHE
2370 DATA FURNISH,PROVIDE,DIVEST
2375 DATA GAIN,REACH,LOSE
2380 DATA GARNISH,ADORN,DEBASE
2385 DATA GENTLE,TAME,FIERCE
2390 DATA GENEROUS,UNSELFISH,GREEDY
2395 DATA GIGANTIC,ENORMOUS,SMALL
2400 DATA GLOOM,DARKNESS,LIGHT
2405 DATA GLUM,MOODY,MERRY
2410 DATA GLORY,HONOR,CONTEMPT
2415 DATA GOOD,VIRTUOUS,EVIL
2420 DATA GRACE,BEAUTY,HOMELINESS
2425 DATA GRADUAL,DELIBERATE,SWIFT
2430 DATA GRANT,ALLOT,CONFISCATE
2435 DATA HAIL,GREET,AVOID
2440 DATA HAPPY,CHEERFUL,DEPRESSED
2445 DATA HARASS,TORMENT,COMFORT
2450 DATA HARM,DAMAGE,BENEFIT
2455 DATA HASTEN,HURRY,DELAY
2460 DATA HEAVY,SERIOUS,LIGHT
2465 DATA HESITATE,FALTER,PROCEED
2470 DATA HONOR,ESTEEM,CONTEMPT
2475 DATA HORRID,AWFUL,BEAUTIFUL
2480 DATA HUMBLE,MODEST,PROUD
2485 DATA IDEAL,PERFECT,ACTUAL
2490 DATA IGNORANT,UNLEARNED,EDUCATED
2495 DATA ILLUSION,FANTASY,REALITY
2500 DATA IMMATURE,CHILDISH,MATURE
2505 DATA IMPART,INFORM,HIDE
2510 DATA IMPROVE,HELP,IMPAIR
2515 DATA INCLUDE,CONTAIN,OMIT
2520 DATA INDOLENT,LAZY,ALERT
2525 DATA INSULT,WRONG,PRAISE
2530 DATA IRRITATE,ANNOY,GRATIFY
2535 DATA JARGON,DIALECT,BABBLE
2540 DATA JESTER,CLOWN,SCHOLAR
2545 DATA JOYOUS,HILARIOUS,SORROWFUL
2550 DATA JUMBLE,CONFUSION,ORDER
2555 DATA JUST,FAIR,DISHONEST
2560 DATA JEER,MOCK,PRAISE
2565 DATA JUSTIFY,EXCUSE,BLAME
2570 DATA KEEN,SHARP,BLUNT
2575 DATA KILL,SLAY,SAVE
2580 DATA KINDRED,FAMILY,STRANGERS
2585 DATA KISS,CARESS,SPURN
2590 DATA KNACK,SKILL,INABILITY
2595 DATA KNOW,COMPREHEND,DOUBT
2600 DATA LABOR,ENDEAVOR,LEISURE
2605 DATA LANGUAGE,SPEECH,GIBBERISH
2610 DATA LATENT,HIDDEN,VISIBLE
2615 DATA LEAN,DEPEND,RAISE
2620 DATA LESSEN,DECREASE,EXPAND
2625 DATA LIBEL,SLANDER,APPLAUSE
2630 DATA LIVELY,ACTIVE,LISTLESS
2635 DATA LOGICAL,VALID,SPURIOUS
2640 DATA LUSCIOUS,DELECTABLE,UNSAVORY

2645 DATA MADNESS,INSANITY,SANITY
2650 DATA MALICE,GRUDGE,AFFECTION
2655 DATA MANY,NUMEROUS,FEW
2660 DATA MANAGE,CONTROL,ABANDON
2665 DATA MEASURE,TEST,GUESS
2670 DATA MEND,FIX,HURT
2675 DATA METTLE,BRAVERY,FEAR
2680 DATA MIGHTY,STRONG,FRAGILE
2685 DATA NAIVE,INNOCENT,WORLDLY
2690 DATA NARROW,BIGOTED,TOLERANT
```

lexicon cont

```
2695 DATA NOTIFY,INSTRUCT,DELUDE
2700 DATA NOBLE,EMINENT,MEAN
2705 DATA NEAT,TRIM,DIRTY
2710 DATA NURTURE,VALUE,REJECT
2715 DATA OBEDIENT,COMPLIANT,OBSTINATE

2720 DATA OBLIGE,COMPEL,PERSUADE
2725 DATA OBSCURE,VAGUE,CLEAR
2730 DATA OBTAIN,ACQUIRE,LOSE
2735 DATA ODIOUS,MEAN,DECENT
2740 DATA OPEN,UNLOCK,CLOSE
2745 DATA PACIFY,CALM,AROUSE
2750 DATA PANIC,ALARM,CALMNESS
2755 DATA PART,PIECE,WHOLE
2760 DATA PERIL,DANGER,SAFETY
2765 DATA PERSIST,LAST,QUIT
2770 DATA PLEAD,BEG,DENY
2775 DATA POLLUTE,CORRUPT,PURIFY
2780 DATA PROPEL,MOVE,STOP
2785 DATA QUAINT,ODD,ORDINARY
2790 DATA QUICK,FAST,SLOW
2795 DATA QUIT,STOP,CONTINUE
2800 DATA QUESTION,ASK,ANSWER
2805 DATA RACKET,NOISE,QUIET
2810 DATA RAPID,FAST,SLOW
2815 DATA RAVAGE,DESTROY,CONSERVE
2820 DATA REACT,REPLY,IGNORE
2825 DATA RUSH,HURRY,DELAY
2830 DATA RUSTIC,RURAL,URBANE
2835 DATA SALARY,WAGES,GIFT
2840 DATA SEIZE,RESTRAIN,RELEASE
2845 DATA SELECT,CHOOSE,REFUSE
     DATA SQUABBLE,ARGUE,AGREE
2855 DATA START,BEGIN,END
2860 DATA STINGY,GREEDY,GENEROUS
2865 DATA TACT,FINESSE,BLUNDER
2870 DATA TANGIBLE,PHYSICAL,SPIRITUAL
2875 DATA TENACITY,PERSEVERANCE,SLOTH
2880 DATA TERSE,BRIEF,LENGTHY
2885 DATA THWART,FOIL,FULFILL
2890 DATA ULTIMATE,FINAL,FIRST
2895 DATA UNCERTAIN,HAZY,CLEAR
2900 DATA UNDER,BELOW,ABOVE
2905 DATA UNIFORM,ORDERLY,ERRATIC
2910 DATA UNTAMED,SAVAGE,GENTLE
2915 DATA USUAL,COMMON,RARE
2920 DATA VACANT,EMPTY,FULL
2925 DATA VAGUE,HAZY,CLEAR
2930 DATA VALID,LEGAL,VOID
2935 DATA VERBAL,SPOKEN,WRITTEN
2940 DATA VULGAR,GROSS,REFINED
2945 DATA WAGES,PAY,GIFT
2950 DATA WAIVE,RENOUNCE,UPHOLD
2955 DATA WRANGLE,QUARREL,HARMONY
2960 DATA WRONG,FALSE,RIGHT
2965 DATA WRECK,DESTROY,MAKE
2970 DATA YEARNING,CRAVING,AVERSION
2975 DATA YIELD,GRANT,REFUSE
2980 DATA YOUTHFUL,YOUNG,OLD
2985 DATA ZEAL,ARDOR,APATHY
2990 DATA ZEALOUS,EAGER,COOL
2995 DATA ZENITH,SUMMIT,DEPTH
```

BASIC 8K RAM

```
0 REM SAVE"D:BASIC8K.RAM"
10 DATA 104,104,133,206,104,133,205,10
4,133,204,104,133,203,104,104,133
14 DATA 207,165,203,41,63,74,133,204,1
69,0,106,133,203,173,1,211,9
18 DATA 2,141,1,211,165,204,24,105,160
,133,204,160,127,165,207,208
22 DATA 10,177,205,145,203,136,16,249,
240,10,234,177,203,145,205,136
26 DATA 16,249,173,1,211,41,253,141,1,
211,96,-1
80 REM 1024 is the start of the casset
te buffer. If you use cassette & have
used page 6, then lower RAMTOP to
85 REM 159 and let A=256*159
90 A=1024:I=A:REM The -1 above is not
part of code.  It just stops the loop
100 READ X:IF X)=0 THEN POKE I,X:I=I+1
:GOTO 100
200 DIM X$(128),A$(128):A$=CHR$(0):A$(
128)=CHR$(0):A$(2)=A$:REM A$ can be wh
atever you want. I used CHR$(0),
205 REM hearts, as a demo point only.
210 ? "ENTER A STRING TO BE SAVED..."
220 INPUT X$:IF X$="" THEN 400
230 IF LEN(X$)<128 THEN X$(LEN(X$)+1)=
A$
240 ? "ENTER ID# OF RAM SECTOR TO WRIT
E (0-63) :;"
250 INPUT IDNO
260 IF IDNO<0 OR IDNO)63 OR INT(IDNO)(
)IDNO THEN GRAPHICS 0:? :? :GOTO 240
270 X=USR(A,ADR(X$),IDNO,0)
280 GOTO 210
400 ? "ENTER ID# OF RAM SECTOR TO READ
 (0-63) ";
410 INPUT IDNO
420 IF IDNO<0 OR IDNO)63 OR INT(IDNO)(
)IDNO THEN GRAPHICS 0:? :? :GOTO 240
425 X$(128)=" ":REM Must be used or pr
ogram will not work.
430 X=USR(A,ADR(X$),IDNO,1)
440 ? X$
490 GOTO 400
500 REM To put program into a string,
change line 90 to read I=1:DIM F$(76),

505 REM Change line 100 to read READ X
: IF X)=0 THEN F$(I)=CHR$(X):I=I+1:G.1
00.
510 REM Change lines 270 and 430 from
A to ADR(F$).
```

```
100 REM BASIC FIX - USE WITH DOS 2.0S
110 OPEN #1,8,0,"D:AUTORUN.SYS":FOR I=
1 TO 183:READ J:PUT #1,J:NEXT I
120 D.255,255,43,155,215,155,165,10,14
1,179,155,165,11,141,188,155
130 D.165,12,141,66,155,165,13,141,67,
155,208,3,32,64,21,162,0,169
140 D.12,157,66,3,32,86,228,169,144,13
3,106,169,3,157,66,3,169,213
150 D.157,68,3,169,155,157,69,3,169,12
,157,74,3,32,86,228,169,156
160 D.133,213,169,188,133,225,160,0,13
2,212,132,224,162,4,177,212
170 D.145,224,200,208,249,230,213,230,
225,202,208,242,230,6,169
180 D.178,133,10,169,155,133,11,169,65
,133,12,169,155,133,13,169
190 REM NEXT BYTE IS LEFT MARGIN (2)
200 D.2,133,82,169
210 REM NEXT BYTE IS RIGHT MARGIN (39)

220 D.39,133,83,169
230 REM NEXT BYTE IS LETTER LUMINANCE
240 D.202,141,197,2,169
250 REM NEXT BYTE IS FOREGROUND COLOR
260 D.148,141,198,2,169
270 REM NEXT BYTE IS BORDER COLOR
280 D.0,141,200,2,24,96,32,159,23,169,
255,133,8,48,20,169,64
290 D.141,66,155,169,21,141,67,155,169
,159,141,179,155,169,23
300 D.141,180,155,32,137,155,76,0,160,
69,58,155,0,156,255,187
310 FOR I=1536 TO 1562:READ J:POKE I,J
:NEXT I
320 POKE 2606,0:I=USR(1536,48128,1024)
:I=USR(1536,40960,7168):POKE 2606,2
330 D.104,162,16,169,11,157,66,3,104,1
57,69,3,104,157,68,3,104,157,73,3,104,
157,72,3,76,86,228
340 D.FOR I=1 TO 73:READ J:PUT #1,J:NE
XT I:CLOSE #1:END
350 D.87,172,87,172,220,100,172,101,17
2,220,220,76,169,78,169,76
360 D.160,191,173,172,175,172,76,172,1
91,130,156,132,156,76,181,191
370 D.160,159,190,159,240,3,76,84,169,
202,208,1,96,76,84,169
380 D.240,4,73,128,133,212,76,186,171,
32,81,218,169,7,133,192
390 D.76,134,188,226,2,227,2,43,155
```

REVERSER by John Kelly

```
1 REM ********************************
2 REM ** FILE COPY *** FILE TAPE   **
3 REM **   REVERSER   3/12/83       **
4 REM **    JOHN R. KELLEY          **
5 REM **   608 S.E. 28TH AVE        **
6 REM **   PORTLAND, OR 97214       **
7 REM ********************************
40 DIM NAME$(20),A$(10),XX$(1)
60 DIM ELIST$(9),MYLIST$(9),RLIST$(9)
70 NAME$="REVERSE"
80 GOSUB 290
90 OPEN #1,4,0,"K:"
100 ? "Welcome to the game of REVERSE"

110 ? "Do you need instructions (Y/N)"
;:GOSUB 870:IF XX=41 THEN GOSUB 1000
120 ? "%How many digits would you":? "
like (3-9)";
130 GOSUB 870:MAX=XX
140 ? XX;
150 IF MAX<3 OR MAX)9 THEN ? "%Sorry..
.only numbers between 3-9":GOTO 120
160 GOSUB 420
170 TURN=TURN+1
180 ? "%"
190 ? "TURN # ";TURN
200 ? :? "Reverse how many? ";
210 GOSUB 870
220 R=XX:? R;
230 FOR WW=1 TO 200:NEXT WW
240 IF R<2 OR R)MAX THEN 270
250 GOSUB 620
260 GOTO 170
270 ? "Can only reverse from 2 - ";MAX
;;" digits":FOR WW=1 TO 1200:NEXT WW
280 GOTO 180
290 X=3
300 GRAPHICS 2
305 POKE 710,0
310 FOR I=1 TO LEN(NAME$)
320 POSITION X+1,3
330 ? #6;NAME$(LEN(NAME$)-I+1,LEN(NAME
$))
340 SOUND 0,INT(RND(0)*255)+1,10,12
350 FOR WW=1 TO 80:NEXT WW
360 X=X+1
370 NEXT I
380 SOUND 0,0,0,0
390 POSITION 9,5:? #6;"by"
400 POSITION 4,7:? #6;"john r  kelley"

410 RETURN
420 FOR K=1 TO MAX
430 MYLIST$(K,K)="0":ELIST$(K,K)="0"
440 NEXT K
450 FOR I=1 TO MAX
460 RN=INT(RND(0)*MAX)+1
470 IF ELIST$(RN,RN)()"0" THEN 460
480 MYLIST$(I,I)=STR$(RN):ELIST$(RN,RN
)=STR$(RN)
490 NEXT I
500 GRAPHICS 2
505 POKE 710,0
510 X=9+INT(MAX/2)
520 FOR I=1 TO MAX
530 GRAPHICS 2:POKE 710,0
540 POSITION X,4
550 ? #6;MYLIST$(1,I)
560 SOUND 1,INT(RND(0)*100)+(2*I),10,1
0:WW=5:GOSUB 690
570 SOUND 1,0,10,10:WW=5:GOSUB 690
580 X=X-1
590 NEXT I
600 SOUND 1,0,0,0
610 RETURN
620 FOR S=1 TO R
630 RLIST$(R-S+1,R-S+1)=MYLIST$(S,S)
640 NEXT S
650 MYLIST$(1,R)=RLIST$(1,R)
660 IF MYLIST$=ELIST$ THEN 720
670 GOSUB 500
680 RETURN
690 REM
700 FOR WAIT=1 TO WW:NEXT WAIT
710 RETURN
720 FOR I=1 TO 9
730 GRAPHICS 2:POKE 710,0
735 POSITION 5,4:? #6;MYLIST$
740 SETCOLOR 4,INT(RND(0)*16),8
750 FOR Q=15 TO 30 STEP 0.5
760 SOUND 0,Q,10,10
770 NEXT Q:SOUND 0,0,0,0
780 NEXT I
790 ? "%You won in ";TURN;" turn";:IF
TURN()1 THEN ? "s";
800 ? "!":SETCOLOR 4,0,0
810 TRAP 840:? "Would you like to play
 again (Y/N)";
820 INPUT A$:IF A$(1,1)="N" THEN 840
830 TURN=0:POP :? "%":GOTO 120
840 ? "Thanks FOR playing!":FOR I=1 TO
 333:NEXT I:GRAPHICS 0:END
870 REM
880 SOUND 0,200,10,8:FOR W5=1 TO 100
890 NEXT W5:SOUND 0,0,0,0:POKE 764,255

900 GET #1,XX:XX=XX-48
910 TRAP 920
920 TRAP 34567
930 RETURN
1000 POKE 752,1:? "%You will be given
some scrambled":? "numbers from 1 to t
he number you"
1010 ? "choose. ( 3-9 ) Hit RETURN":GO
SUB 870
1020 ? "%By reversing blocks of number
s try":? "to unscramble them so they a
re"
1030 ? "in the usual order(1-9) Hit RE
TURN";:GOSUB 870
1040 ? "%When you reverse a block of n
umbers":? "The reversal always starts
from the"
1050 ? "left. For example..Hit RETURN"
:GOSUB 870
1060 ? "%if you reverse 3 numbers of t
his":? "combination: 6421735 you would
 get"
1070 ? "2461735. Only the 6, 4, and 2
are":? "reversed.  Ready? Hit RETURN";
:GOSUB 870:RETURN
```

ACTION

```
BYTE FUNC INDEX(CARD A1,A2)
;
; RETURNS THE POSITION OF STRING '
; IN S1, RETURNS 0 IF FAILURE
;
; A1,A2 ADDRESS OF S1,S2 RESPECTIVELY
;
BYTE ARRAY S1,S2
BYTE I,J,TEMP
 S1=A1
 S2=A2
 IF S1(0)>=S2(0) THEN
  FOR I=1 TO S1(0)-S2(0)+1
  DO
   FOR J=1 TO S2(0)
   DO
    IF S1(I+J-1)=S2(J) THEN
     IF J=S2(0) THEN
      RETURN(I)
     FI
    ELSE
     EXIT
    FI
   OD
  OD
 FI
RETURN (0)
```

X Handler

```
    1 X:HANDLER
7 DIM F$(15)
10 GRAPHICS 0:? "X: creator":? :? "che
cking data..."
15 LINE=1000:TRAP 90
20 FOR X=1 TO 10:READ BYTE
30 TOT=TOT+BYTE
40 IF TOT>999 THEN TOT=TOT-1000
50 NEXT X:READ CHKSUM
60 IF TOT<>CHKSUM THEN ? "data error i
n line ";LINE:END
70 LINE=LINE+10:GOTO 20
90 IF LINE<1300 THEN ? "missing a data
 line":END
100 ? "Enter DEV:FILENAME";:INPUT F$
105 ? "creating file..."
110 RESTORE 1000
120 OPEN #1,8,0,F$
130 TRAP 200
140 FOR X=1 TO 10:READ BYTE
150 PUT #1,BYTE:NEXT X
160 READ CHKSUM:GOTO 140
200 IF PEEK(195)<>6 THEN ? "Error ";PE
EK(195):END
210 CLOSE #1:? "Awesome...file complet
ed!":END
1000 DATA 255,255,0,34,251,34,165,12,1
41,60,207
1010 DATA 34,165,13,141,61,34,169,59,1
33,12,28
     DATA 169,34,133,13,169,31,141,231
,2,169,120
1030 DATA 35,141,232,2,160,0,185,26,3,
240,144
1040 DATA 6,200,200,200,192,34,144,244
,96,169,631
1050 DATA 88,153,26,3,169,66,153,27,3,
169,488
1060 DATA 34,153,28,3,96,32,65,34,32,1
0,975
1070 DATA 34,96,77,34,97,34,119,34,132
,34,666
1080 DATA 150,34,174,34,189,74,3,201,9
,240,774
1090 DATA 86,169,0,141,26,35,169,192,1
41,27,760
1100 DATA 35,76,171,34,189,74,3,41,8,2
40,631
1110 DATA 66,173,26,35,141,28,35,173,2
7,35,370
1120 DATA 141,29,35,76,171,34,32,151,3
4,176,249
1130 DATA 42,32,233,34,177,224,76,255,
34,172,528
1140 DATA 27,35,240,29,72,32,216,34,10
4,32,349
1150 DATA 233,34,145,224,76,255,34,172
,27,35,584
1160 DATA 32,216,34,204,29,35,208,9,20
5,28,584
1170 DATA 35,208,4,160,136,56,96,160,1
,24,464
1180 DATA 96,189,66,3,201,37,208,16,18
9,77,546
1190 DATA 3,201,192,144,24,141,27,35,1
89,76,578
1200 DATA 3,141,26,35,173,26,35,157,76
,3,253
1210 DATA 173,27,35,157,77,3,76,171,34
,160,166
1220 DATA 171,96,173,26,35,133,224,192
,208,208,632
1230 DATA 5,160,216,140,27,35,132,225,
96,172,840
1240 DATA 1,211,140,30,35,160,0,140,14
,210,781
1250 DATA 140,14,212,160,252,140,1,211
,252,34,197
1260 DATA 30,35,160,0,96,172,30,35,140
,1,896
1270 DATA 211,160,64,140,14,212,164,16
,140,14,31
1280 DATA 210,238,26,35,208,3,238,27,3
5,160,211
1290 DATA 1,96,0,0,0,0,0,224,2,225,759

1300 DATA 2,0,34,0,0,0,0,0,0,0,795
1310 REM * 310 BYTES
```

ACTION

```
; DEMO TO SHOW HOW TO EMBED CONTROL
; CHARACTERS IN ACTION! TO USE THE
; ACTION! EDITOR AS A WORD PROCESSOR
; FOR AN EPSON MX-80 PRINTER
;       by John Logan


*

E E G_EMBEDDED CONTROL KEYS

-------------------------------------

_
E F E H
TEST START
_CTRL N ENLARGES ONE LINE
TEST LINE
.CTRL O SHRINKS
TEST LINE
-CTRL R CANCELS
TEST LINE
E E ESC ESC E EMPHASIZES
TEST LINE
E F ESC ESC F CANCELS
TEST LINE
E G ESC ESC G DOUBLE STRIKES
TEST LINE
E H ESC ESC H CANCELS
TEST LINE
E E E G ESC ESC E ESC ESC G DOES BOTH
TEST LINE
E F E H ESC ESC F ESC ESC H CANCELS BOTH


TEST LINE
E 4 ESC ESC 4 ITALICS ON
TEST LINE
E 5 ESC ESC 5 ITALICS OFF
TEST LINE



TEST LINE
.CTRL I INDENTS
NEXT LINE



CTRL L FORM FEEDS TO NEXT PAGE
ESC ESC A CTRL A TO CTRL Z SETS LINE F
EED TO 1 TO 26 /72

ESC ESC @ CANCELS EVERYTHING
```

```
TO PRINTOUT
PRINT [TYPE THE TITLE]
MAKE "TITLE RL
SETWRITE "P:
PR :TITLE
PR []
SETWRITE []
PRINT [TYPE YOUR NAME AND THE DATE]
MAKE "NAME RL
SETWRITE "P:
PR :NAME
PR []
SETWRITE []
PRINT [TYPE THE NAME OF THE PROCEDURE]

MAKE "PRO RL
SETWRITE "P:
PO :PRO
SETWRITE []
END
```

THREE ACTION PROGRAMS

by JOHN LOGAN

PRINTER ACT 012
CTRLKEYSACT 005
NAVAJO ACT 018

PRINTER.ACT documents how to enter EPSON printer codes in an ACTION! program with examples of each feature.

CTRLKEYS.ACT shows how to embed control characters in an ACTION! source document. This allows one to use the ACTION! text editor as a word processor with printer control.

NAVAJO.ACT draws a bargello needle point, a patchwork quilt and an abstract mountain in a Navajo style. Push the joystick trigger to change patterns. Push the stick left & right to select color register; push it up & down to change color value.

```
; PRINTER CODES IN ACTION
; by John Logan
;
; ACE Newsletter
; 3662 Vine Maple Dr
; Eugene, OR 97405
;    5i4 year
;   July 1985
;
PROC PRINTER()
OPEN (2,"P:",8,0)


PUTD(2,27) PUTD(2,'E)
PUTD(2,14)
PRINTDE(2,"PRINTER CODES ")
PRINTDE(2,"_____")
PUTD(2,27) PUTD(2,'F)

;  Rem SET LINE FEED
PUTD(2,27) PUTD(2,'3) PUTD(2,70)

PRINTDE(2,"")
PRINTDE(2,"STANDARD PRINT")

PRINTDE(2,"EMPHASIZED STD  PUTD(2,27)
PUTD(2,'E)")
PUTD(2,27) PUTD(2,'E) PRINTDE(2,"EMPHA
SIZED STD")
PRINTDE(2,"TO CANCEL PUTD(2,27) PUTD(2
,'F) ")
PUTD(2,27) PUTD(2,'F) PRINTDE(2,"CANCE
LLED")

PRINTDE(2,"DOUBLE STRIKE PUTD(2,27) PU
TD(2,'G) ")
PUTD(2,27) PUTD(2,'G) PRINTDE(2,"DOUBL
E STRIKE STD")
PRINTDE(2,"TO CANCEL PUTD(2,27) PUTD(2
,'H)")
PUTD(2,27) PUTD(2,'H) PRINTDE(2,"CANCE
LLED")

PRINTDE(2,"ENLARGED PRINT PUTD(2,14)")

PUTD(2,14) PRINTDE(2,"ENLARGED PRINT "
)
PRINTDE(2, "CANCELS AUTOMATICALLY AFTE
R EA LINE FEED")

PRINTDE(2, "CONDENSED PRINT PUTD(2,15)
")
PUTD(2,15) PRINTDE(2,"CONDENSED PRINT
")
PRINTDE(2, "TO CANCEL, PUTD(2,18)")

PUTD(2,18) PRINTDE(2,"CANCELLED")
PRINTDE(2,"") PRINTDE(2,"")

PUTD(2,14) PRINTDE(2,"COMBINATIONS:")
PUTD(2,14) PRINTDE(2,"STD ENLARGED")

PUTD(2,27) PUTD(2,'E)
PUTD(2,14) PRINTDE(2,"ENLARGED + EMPHA
SIZED")
PUTD(2,27) PUTD(2,'F)

PUTD(2,27) PUTD(2,'G)
PUTD(2,14) PRINTDE(2,"ENLARGED + DOUBL
E")

PUTD(2,27) PUTD(2,'E)
PUTD(2,14) PRINTDE(2,"ENLARGED + DOUBL
E + EMPHASIZED")

; STD LINE FEED
PUTD(2,27) PUTD(2,'2)

CLOSE(2)

RETURN
```

USING THE UNUSED

Here are a couple of recent articles describing how to use some of that extra memory in the XL (and XE?) machines. — J.B.

X:HANDLER

Using the XL's Extra Memory As a Device
[Reprint: MACE, June, 1985]

The Atari XL computers have an extra 16k of RAM hidden beneath their ROM. This is why the XLs boast 64k RAM while the 800s have only 48k. However, this extra memory almost always goes to waste. There are many new products coming out soon which will use it, but currently there are very few programs besides the Translator Disk and DOS XL [and Flight Simulator II — ed.] which utilize it; most don't even know about it. In BASIC and all other languages the XLs still have the same amount of free memory as a 48k 800. What good is the extra memory if it is not used?

This program allows you to access 14k of the extra 16k as device X: (2k is always allotted to I/O space). You can do most things with it which you can do with any other device: You can SAVE "X:", LOAD "X:", PRINT to a channel opened to X:, GET from a channel opened to X:, etc. You can also NOTE and POINT as with a disk drive. The X: device will not be killed by RESET. Anything saved to it will be completely invisible to everything except the X: handler itself. Do not use X: when the DOSXL:XL file of DOS XL is in use, or DOS will be clobbered.

To use the program, type in the BASIC listing. It will ask you for a filespec (D:AUTORUN.SYS or C:) and then create the file . . . After rebooting the computer or loading the file from DOS, you will have device X: at your disposal. From BASIC try LIST "X:" when there is a program in memory. The screen will flicker as the ROM character set is switched on and off. This flicker serves the same purpose as the beeping of the disk drive. Type NEW and then LIST the program to verify it is gone. Now type ENTER "X:" and LIST again. Ta da! It's back. If the X: handler should lock up the computer and leave a bunch of squiggly lines on the screen, press RESET and try the operation again. Check to make sure you typed in the program correctly.

HOW IT WORKS

CIO, the Central Input/Output utility in the operating system, organized I/O by devices. Each device has its own handler (or driver), which is a program which has the routines necessary to communicate with that device. There are five device handlers resident in the OS ROM: the screen editor, E:; the display handler, S:; the keyboard handler, K:; the printer handler, P:; and the cassette handler, C:. Each has a vector table with pointers to five routines: OPEN the device; CLOSE; GET a byte from the 'evice; PUT; get the STATUS of the device; and do a device-dependent r special XIO command. Some of the routines in each handler are not used, because you cannot input (OPEN #chan,4,...) from the printer, or output (OPEN #chan,8,...) to the keyboard. The five resident handlers have their vector tables in order in ROM starting at $E400 (58368), each having 16 bytes allotted to it. There are the six vectors of two bytes each, a JMP to the power-up initialization of the device, and a spare byte. CIO keeps track of where the vector tables for the handlers are by keeping a handler address table, HATABS, which starts at $31A (794). Each entry in the table takes up three bytes: the designation letter (E, S, K, etc.), nd the two-byte address of the handler's vector table. Additional entries can be added to the table; the D: handler is added upon booting DOS, and the X: handler is added upon loading this program. Whenever I/O is done, a call is made to CIO, which tracks down the device handler and takes care of everything.

At the beginning of the program is a routine which makes X: RESET-proof. It steals the DOS re-init vector and changes it to point to the X: re-init routine, which restores both the D: and X: entries to HATABS after each RESET. Following this is the X: vector table, and then the actual routines. These routines are similar to ones used by Bill Wilkinson for his program which used ordinary memory as a device from the September, 1982 **COMPUTE!** magazine. Near the end of the program there are two routines which manage the OS ROM while bytes are being read or written to the RAM underneath it. Before disabling the ROM to expose the RAM underneath, all interrupts which use the ROM must be disabled, because if an interrupt occurs when there are no ROM interrupt routines the system will crash. Bit zero of PIA chip location $D301 (54017, formerly PORTB in the 400/800) controls the state of the CS ROM, and bit one controls the state of BASIC ROM. There is also a routine which makes sure all reading and writing of data skips over the I/O chip region from $D000 to $D7FF (53248 to 55295).

The X: device is useful as a RAMdisk. It is much faster than floppies and can contain up to 14k of data without giving an error. This is the equivalent of over 100 single density sectors. Random access can be obtained after opening X: by using NOTE and POINT with the low and high bytes of the desired address instead of the sector and byte numbers of a disk. X: can be opened for append by OPEN #chan,9,0,"X:". X: is perfect for temporary, fast storage. Have fun.

— Ken Alexander

HIDDEN XL RAM

(Reprint: HAUG, June, 1985)

This program is a slight modification of one written by Kirt Grittner and printed in the March, 1985 MAAUG Newsletter of Madison, WI users group. It shows how to use the 8,192 bytes of RAM under BASIC to store and retrieve 64 different 128 character strings (8,192 bytes). It's almost like having Microsoft string arrays. The April A.N.A.L.O.G. has a RAM OS program in it which lets you access the 14k of RAM under the operating system. I believe the key to using over 56k of RAM with BASIC, in the 800XL, may well lie within this program and the one in the April ANALOG.

The 76 bytes of machine code are in lines 10 through 26. It is located into the cassette buffer by line 100. Lines 200 through 270 show how to load the 64 X$s. Lines 400 through 490 show how to retrieve any one of the 64 X$s. This will provide FREE space to store the pointer file for 2730 random access records, or whatever else turns you on. Operation of the program is FAST. There is nothing in the screen operation to tell you BASIC (and the OS) was turned off and on. It will NOT work with a 400 or 800. I revised it to make it relocatable, and then used the cassette buffer, as that is used only when doing a coldstart with a disk. Cassette users should use the STRING version. Anything put in a protected area of RAM can be passed from one program to another, therefore your RAM sector data could be passed from one to another program, unless you do a coldstart.

Kurt Grittner was going to tell the Madison, WI users group how to access the rest of the 24k at their next meeting. I hope he and the group are willing to share the procedure with the rest of us.

NEXT

MEETING

WED.

SEPTEMBER

11TH

ENHANCED BASIC

Enhancements to Basic, by First Byte, Box 32, Rices Landing, PA 15357.

Do you ever get the feeling something is missing from Atari BASIC? After all, lots of publication articles are about BASIC automatic numbering or renumbering programs. BASIC english error message programs, BASIC block delete programs, converter programs, and most of all, how to talk to DOS from BASIC. Well, as you probably already know, there's just so much you can put in an 8k ROM cartridge. But worse yet, Atari BASIC sometimes locks up, and Rev. B BASIC has more problems.

This presents the Atari BASIC user with several options. You can buy a "real" computer (not unless it plays Star Raiders and Joust!). You can buy a full function BASIC such as Microsoft BASIC, or BASIC XL from OSS. They are somewhat expensive and no always 100% compatible with Atari BASIC. There are also programs such as Monkey Wrench from Eastern House and some other disk based packages which "enhance" Atari BASIC.

Well, a couple of months ago, I mailed $14.95 off to First Byte for their **Enhancements to Basic** program. ETB is advertized to work on all Ataris with a minimum of 48k of RAM and revision A, B, or C Atari BASIC. I really wasn't expecting much for $14.95, but hoped for something which would eliminate the half dozen BASIC utilities I'm always loading when entering and debugging BASIC programs. What I received far exceeded my expectations.

First of all there was a letter of apology from the president of First Byte for the delay in shipment. It seems they decided to revise the program so it would be compatible with several Atari DOS's (Atari 2.0, 2.5, 3.0, OSS DOS XL, SpartaDOS).

The program creates a boot disk which customizes itself to your particular flavor of Atari computer hardware and version of BASIC. It then generates a disk based version of BASIC. This means 400/800 owners no longer need to insert the cartridge. XL/XE owners **do need** to boot with the OPTION key held down. First Byte also claims to correct the documented causes of 800 and 800 XL system lockup and the 16 byte memory expansion bug of Rev. B BASIC in the 800 XL. This alone should make the deal attractive to the XL folks with Rev. B BASIC who think it's ridiculous to have to send another $15 to Atari to get a BASIC which works (maybe).

Cold start boot displays a screen with your name, address, and serial number (how do they customize this thing for $14.95?), and a title bar across the top of the screen. The title bar indicates the type of computer on which the program was initialized (800 or XL/XE) and the type of DOS the program recognized at boot time. A menu then allows selecting three levels of enhancement: Half, Full, and Full with Trace. These modes use approximately 4k for half mode and 8k for full and full trace modes. Thus, the less the enhancement, the more memory for your program. It should be interesting to see if First Byte can modify the program to make use of the expanded memory in the new Atari XE.

I first appreciated the display of error messages in english instead of cryptic codes. No more looking for the book. The next nice touch is the availability of a HELP screen. **ETB** then adds the following immediate mode commands to BASIC: Automatic line numbering and renumbering, delete a block of lines, display disk directory, rename disk files, lock and unlock disk files, format disk in single or double density, and list or change the variable names used in a program.

Other commands provide for DOS access, protecting a BASIC program from editing, sending screen output to the printer, setting screen margins, key board lock, key delay/repeat rates, key click disable, hex to decimal conversion and a binary load file for recovering erased DOS 2.oS files is available.

You can also set up the keys 4,5,6,7,8,9 and 0 in conjunction with the Control or Control plus Shift keys to print strings of characters to the screen with a single key stroke (if I could just get that with AtariWriter!). This is great for printer control codes and repetitious command or string entry.

Last but not least is a program mode trace function. This function sets up an alternate screen to which program lines are printed as they are being executed. Commands are available to switch between screens at any time, allowing you to observe the lines which are the source of program bugs. This is great for debugging nested loops or for just figuring out the flow of that old program anyway. The documentation is relatively complete with about 30 pages of plain english and lots of caution notes to point out the got-yas.

The program does take up memory space, but you are allowed to specify the level of commands you will need to use. Several commands, such as margin width, key click, key repeat rates, etc. would probably be better used if they were definable during the system setup. It would be nice if you could choose screen default colors at setup time also. On an Atari 800, the program crashes (dead screen and keyboard) if you fail to remove the BASIC cartridge. First Byte does give fair warning that ETB is designed to aid in the writing and debugging of BASIC programs, and that it really offers no advantage when simply running utility or game programs.

First Byte offers a 30 day full refund guarantee. When is the last time you heard that on a software package? It's really refreshing to find a company willing to put its money where its mouth is and offer a professional quality program at a very reasonable price.

— Ron Robinsc
Florissant, MO

PAPERCLIP

Review by Mike Dunn, Co-Editor

(Batteries included, 30 Mural St., Richmond Hill, Ontario, L4B 1B5 Canada, $70)

For several months I have been receiving advertizments about this new word-processor for the Atari and how is was as good as or even better than the "big" word-processors such as WordStar, Perfect Writer, etc. Then the results of a contest in Canada among various computers and programs costing up to $20,000, reporting that the Atari version of PaperClip was not only the "Best Buy", but comparible to the bigboys. Other than thinking that Batteries Included should be congratulated for being one of the very few companies coming out with a major new product for the Atari, I thought that they were sending me a lot of "Hype".

I finally received a copy, and can say they do have a real winner, about as good for most uses as the "bigboys".

You begin reading the very clearly written 155 page manual, which first tells you to back up your disk! The program comes with a "key" which plugs into your joystick for protection, allowing you to make back up disks. Similiar to the key from the Synapse program Filemanger 800, which is not too surprising since one of the authors of PaperClib is Dan Moore, the author of SynFile+.

You can then customize your word-processor to your needs, setting many items such as screen margins (with scrolling to a larger than 40 column screen if you like), the size of the two windows, color of screen (I like the black letters on white background), printer features for most any printer, including allowing you to make a custom one with microjustification, Macros, and many other items. You can even use an AutoSave to disk and use the editing arrow keys without the control key if you want. You can then save your options, and they will automatically come on each time you boot up your disk.

When you start typing, the first thing you notice is the beautiful character set with true descenders which are very easy to read.

Besides the usual editing functions whihc all word-processors have, there is an Undo command, Multiple Global Substitution, Tags insert/overwrite toggle, and a Letter Swap toggle allowing you transpose letters as well as a Word Swap Toggle-feature I really can use but have not found on an Atari Word-processor. For students, there is a word count.

There are the usual Printer formatting functions as well as some unusual ones such as MicroJustification, three lines of Headers or Footers or both, and the ability to use a conditional page break (eg. to make sure a table is all on one page). There is also a very nice print preview option. Unlike many other word-processors, the editor options are different from the printer formatting, so you can make both the same or both different. You can use a 40 column screen and an 80 column printer, or an 80 column scrolling screen with an 80 column printer. Double column printing is done by printing both columns at the same time, and you can print to the disk drive for modem use.

There are also some special functons available — a nifty calculator for math functions which prints the answer, ability to make a Table of Contents and User defined commands, are just some of them. Mail merge is easily done, as well as Macros for "boilerplate" (repeating the same phase with a single keystroke).

If all this is not enough for your $50, there are some utility programs thrown in, including a very extensive printer driver maker, and a **graphics dump** allowing you to display and print Koala or Atari Tablet, LightPen, SynTrend, B/Graph, Fun with Art or Atari Paint pictures either with your document or by itself.

What doesn't this program have compared to others? It does not have a spelling checker, you can only get 80 columns on the screen by moving a 40 column window, and when typing very fast (faster than I can type but Jim Bumpas can type 80 words a minute) it seems to lose an occasional character. There are so many commands that it is difficult to remember them, although the prompt line on the bottom helps a lot, as well as the built in Help files. I understand that a spelling program will be availble soon, as well as an 80 column cartridge — then PaperClip will have everything! I have not used the Writer's Tool (O.S.S.) which is Jim's favorite, but PaperClip would be hard to beat.

Atari Computer Enthusiasts

A.C.E. is an independent, non-profit and tax exempt computer club and user's group with no connection to Atari Corp. We are a group interested in educating our members in the use of the Atari Computer and in giving the latest News, Reviews and Rumors.

All our articles, reviews and programs come from you, our members.

Our membership is world-wide; membership fees include the A.C.E. Newsletter. Dues are $14 a year for U.S., and $24 a year Overseas Airmail and include about 10 issues a year of the ACE Newsletter.
Subscription Dep't: 3662 Vine Maple Dr., Eugene, OR 97405.

**President— Dick Barkley, 2907 Wingate, Eugene, OR 97405 503-344-5843

Vice Pres-Larry Gold, 1927 McLean Blvd., Eugene, Or 97405 503-686-1490

Librarians-Ron and Aaron Ness, 374 Blackfoot, Eugene, Or 97404 (503)689-7106.

Editors— Mike Dunn, 3662 Vine Maple Dr., Eugene, Or 97405 503-344-6193

Jim Bumpas, 4405 Dillard Rd., Eugene, Or 97405 503-484-9925

E.R.A.C.E. (Education SIG Editor)-Nora Young, 105 Hansen Lane, Eugene, Or 97404 /503-688-1458

Send 50c stamps or coin ($1 overseas) to the Ness' for the new, updated ACE Library List-new in May 84 !

Bulletin Board
(503) 343-4352

On line 24 hours a day, except for servicing and updating. Consists of a Tara equipped 48K Atari 400 with a TARA keyboard, 2 double-density double sided disk drives, an Epson MX80 printer, a MicroBits 1100E Modem; running the FoReM Bulletin Board software distributed by MicroBits.

Best of ACE books

Volume 1 contains bound issues of the ACE Newsletter from the first issue, Oct 81 to June of 1982

Volume 2 covers July 1982 to June 1983

Only $12 each ($2 extra for Airmail). Available only from:

George Suetsugu
45-602 Apuapu St
Kanoehe, HI 96744

SortFinder 1.2
A composite index of Atari related articles from 5 popular computer periodicals from Apr '81 to June '83, including ACE. Only $6 for ACE member from:

Jim Carr, Valley Soft
2660 S.W. DeArmond
Corvallis, OR 97333

ATARI COMPUTER ENTHUSIASTS
3662 Vine Maple Dr. Eugene OR 97405

FIRST CLASS MAIL